Network Technologies

Semih Şenel
Yıldız Technical University
semihsenel99@gmail.com

Introduction

Network Layer

Responsible for delivering packets between endpoints over multiple links

Application
Transport
Network
Link
Physical

Design Issues

- Store-and-forward packet switching
- Connectionless service - datagrams
- Connection-oriented service - virtual circuits
- Comparison of virtual-circuits and datagrams

Store-and-Forward Packet Switching

The major components of the network are the ISP(Internet Service Provider)'s equipment (routers connected by transmission lines), shown inside the shaded oval, and the customers' equipments, shown outside the oval. H1 is directly connected to one of the ISP's routers, A, perhaps as a home computer that is plugged into a DSL modem. In contrast, H2 is on a LAN, which might be an office Ethernet, with a router, F, owned and operated by the customer. This router has a leased into the ISP's equipment. We have shown F as being outside the oval because it does not belong to the ISP.

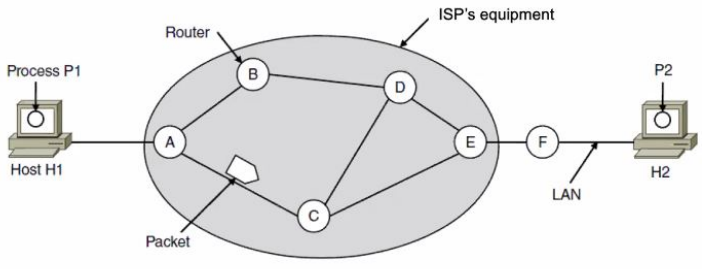


Figure 1: Packet Switching

A host with a packet to sent transmits it to the nearest router, either on its own LAN or over a point-to-point link to the ISP. The packet is stored there until it has fully arrived and the link has finished its processing by verifying the checksum. Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered. This mechanism is store-and-forward packet switching.

Connectionless Service - Datagrams

Having looked at the two classes of service the network layer can provide to its users, it is time to see how this layer works inside. Two different organizations are possible, depending on the type of service offered. If connectionless service is offered, packets are injected into the network individually and routed independently of each other. No advance setup is needed. In this context, the packets are frequently called datagrams (in analogy with telegrams) and the network is called a datagram network.

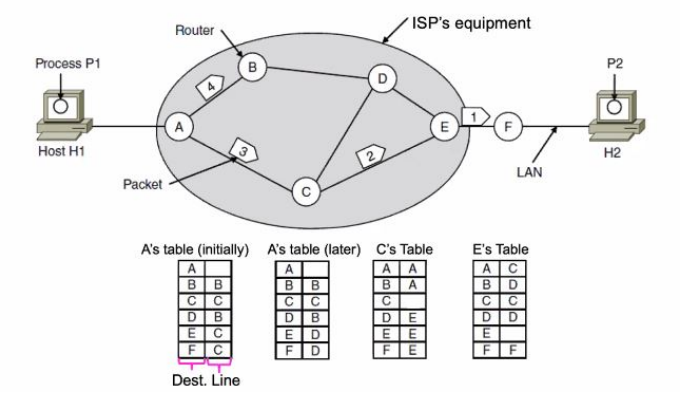


Figure 2: Datagrams

Connection-Oriented - Virtual Circuits

If connection-oriented service is used, a path from the source router all the way to the destination router must be established before any data packets can be sent. This connection is called a VC (virtual circuit), in analogy with the physical circuits set up by the telephone system, and the network is called a virtual-circuit network.

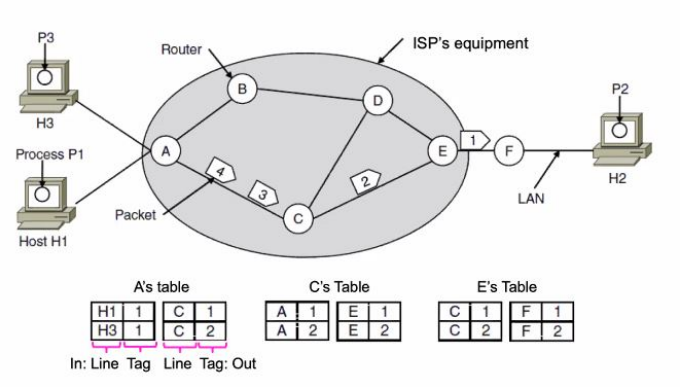


Figure 3: Virtual Circuits

Comparison of Virtual-Circuits & Datagrams

Issue	Datagram network	Virtual-circuit network
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Routing Algorithms

The main function of the network layer is routing packets from the source machine to the destination machine. The algorithms that choose the routes and the data structures that they use are a major area of network layer design.

The **routing algorithm** is the part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on. If the network uses datagrams internally, this decision must be made anew for every arriving data packet since the best route may have changed since last time. If the network uses VCs internally, routing decisions are made only when a new virtual circuit is being set up.

- Model the network as a graph of nodes and links
- Decide what to optimize (e.g., fairness vs efficiency)
- Update routes for changes in topology(e.g., failures)

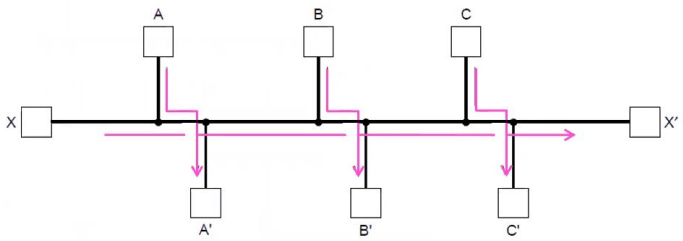


Figure 4: Routing

- Forwarding is the sending of packet along a path.

The Optimality Principle

If router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route. It's called **optimality principle**.

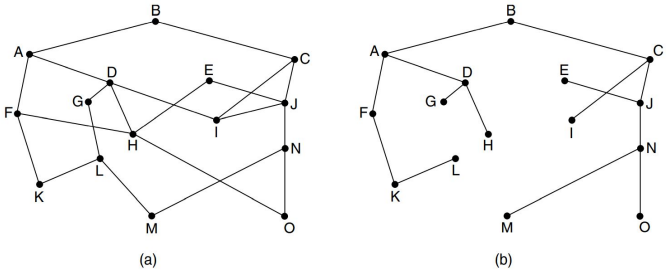


Figure 5: (a) A network. (b) A sink tree for router B.

In other words, each portion of a best path is also a best path; the union of them to a router is a tree called the sink tree.

Shortest Path Algorithm

The idea is the build a graph of the network, with each node of the graph representing a router and each edge of the graph representing a communication line, or link. To choose a route between a given pair of routers, the algorithm just finds the shortest path between them on the graph.

Dijkstra's algorithm computes a sink tree on the graph:

- Each link is assigned a non-negative weigth/distance
- Shortest path is the one with lowest total weight
- Using weights of 1 gives paths with fewest hops

Algorithm:

- Start with sink, set distance at other nodes to infinity
- Relax distance to other nodes
- Pick the lowest distance node, add it to sink tree
- Repeat until all nodes are in the sink tree

Flooding

When a routing algorithm is implemented, each router must make decisions based on local knowledge, not the complete picture of the network. A simple local technique is **flooding**, in which every incoming packet is sent out on every outgoing line except the one it arrived on.

Each node floods a new packet received on an incoming link by sending it out all of the other links.

Nodes need to keep track of flooded packets to stop the flood; even using a hop limit can blow up exponentially.

Distance Vector Routing

Computer networks generally use dynamic routing algorithms that are more complex than flooding, but more efficient because they find shortest paths for the current topology. Two dynamic algorithms in particular, distance vector routing and link state routing, are the most popular.

A **distance vector routing** algorithm operates by having each router maintain a table giving the best known distance to each destination and which link to use to get there. These tables are updated by exchanging information with the neighbors. Eventually, every router knows the best link to reach each destination.

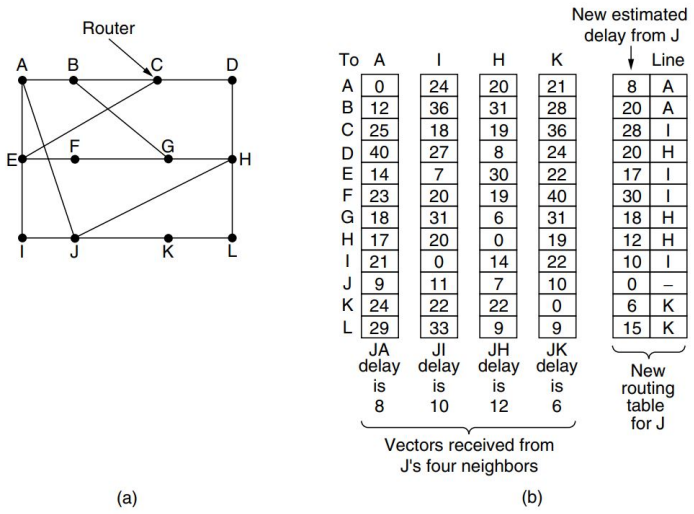


Figure 6: (a) A network, (b) Input from A, I, H, K, and the new routing table for J.

- **The Count-to-Infinity Problem**

The settling of routes to best paths across the network is called **convergence**. Distance vector routing is useful as a simple technique by which routers can collectively compute shortest paths, but it has a serious drawback in practice: although it converges to the correct answer, it may do so slowly.

Failures can cause DV to "count to infinity" while seeking a path to an unreachable node.

Figure 7: The count-to-infinity problem.

Link State Routing

Distance vector routing was used in the ARPANET until 1979, when it was replaced by link state routing. The primary problem that caused its demise was that the algorithm often took too long to converge after the network topology changed (due to the count-to-infinity problem). Consequently, it was replaced by an entirely new algorithm, now called **link state routing.**

The idea behind link state routing is fairly simple and can be stated as five parts. Each router must do the following things to make it work.

1. Discover its neighbors and learn their network addresses.
2. Set the distance or cost metric to each of its neighbors.
3. Construct a packet telling all it has just learned.

4. Send this packet to and receive packets from all other routers.
5. Compute the shortest path to every other router.

Link State Routing - LSPs

- LSP (Link State Packet) for a node lists neighbors and weights of links to reach them.

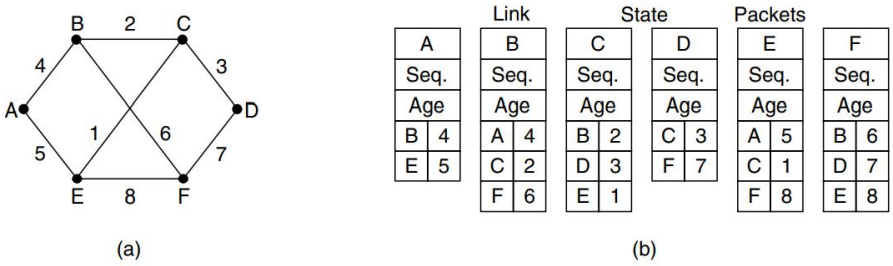


Figure 8: (a) A network. (b) The link state packets for this network.

Link State Routing - Reliable Flooding

Seq. number and age are used for reliable flooding

- New LSPs are acknowledged on the lines they are received and sent on all other lines
- Example shows the LSP database at router B

			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

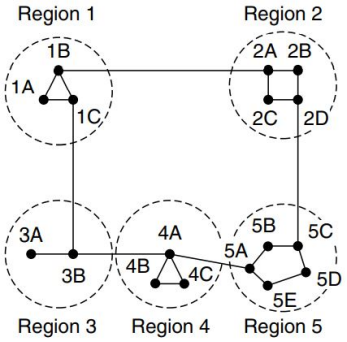
Figure 9: The packet buffer for router B

Hierarchical Routing

As networks grow in size, the router routing tables grow proportionally. Not only is router memory consumed by ever-increasing tables, but more CPU time is needed to scan them and more bandwidth is needed to send status reports about them.

When hierarchical routing is used, the routers are divided into what we will call **regions**. Each router knows all the details about how to route packets to destinations within its own region but knows nothing about the internal structure of other regions.

Hierarchical routing reduces the work of route computation but may result in slightly longer paths than flat routing.



(a)

Full table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

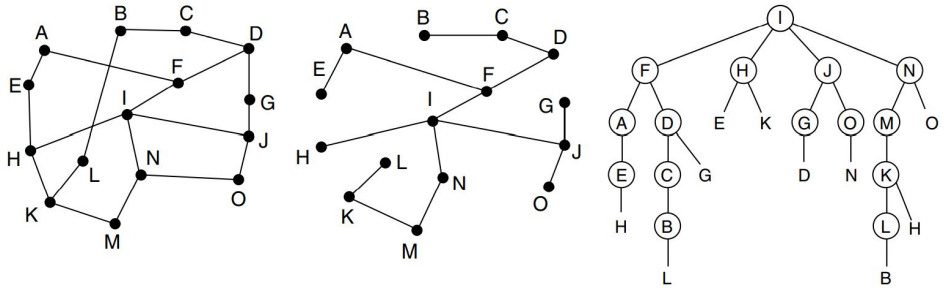
Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

Figure 10: Hierarchical Routing

Broadcast Routing

In some applications, hosts need to send messages to many or all other hosts. From example, a service distributing weather reports, stock market updates, or live radio programs might work best by sending to all machines and letting those that are interested read the data. Sending a packet to all destinations simultaneously is called **broadcasting**. Various methods have been proposed for doing it.



(a) (b) (c)

Figure 11: Reverse path forwarding. (a) A network. (b) A sink tree. (c) The tree built by reverse path forwarding.

- RPF (Reverse Path Forwarding): Send broadcast received on the link to the source out all remaining links.
- Alternatively, can build and use sink trees at all nodes.

Multicast Routing

Sending a message to such a group is called **multicasting**, and the routing algorithm used is called **multicast routing**. All multicasting schemes require some way to create and destroy groups and to identify which routers are members of a group.

Multicast Routing - Dense Case

- Uses a different tree for each group and source

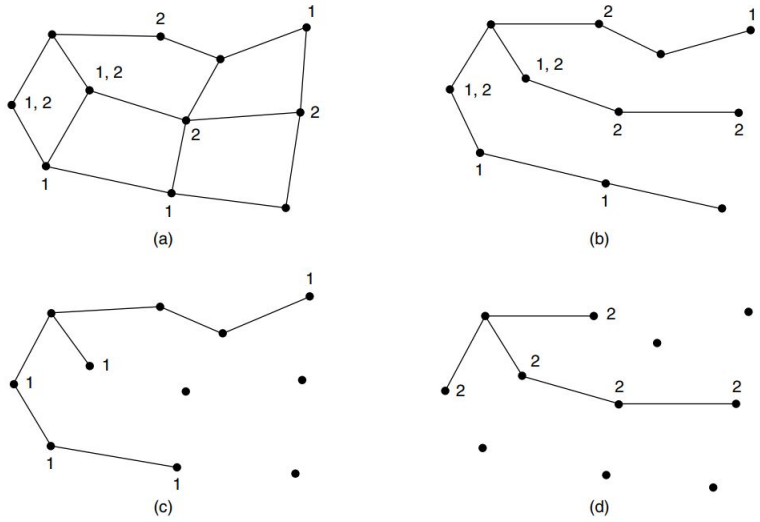


Figure 12: (a) A network. (b) A spanning tree for the leftmost router. (c) A multicast tree for group 1. (d) A multicast tree for group 2.

Multicast Routing - Sparse Case

CBT (Core-Based Tree) uses a single tree to multicast

- Tree is the sink tree from core node to group members
- Multicast heads to the core until it reaches the CBT

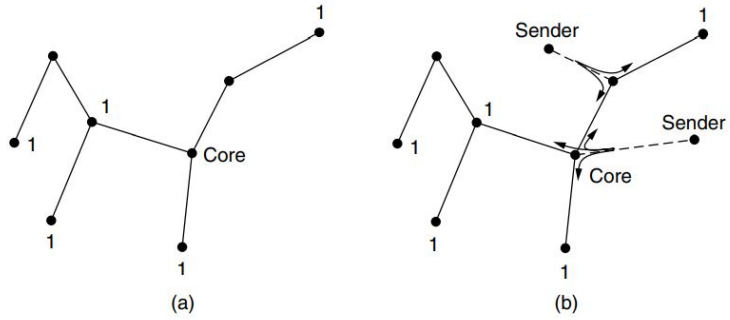


Figure 13: (a) Core-based tree for group 1. (b) Sending to group 1.

Anycast Routing

In anycast, a packet is delivered to the nearest member of a group. Schemes that find these path are called **anycast routing**.

Sometimes nodes provide a service, such as time of day or content distribution for which it is getting the right information all that matters, not the node that is contacted; any node will do.

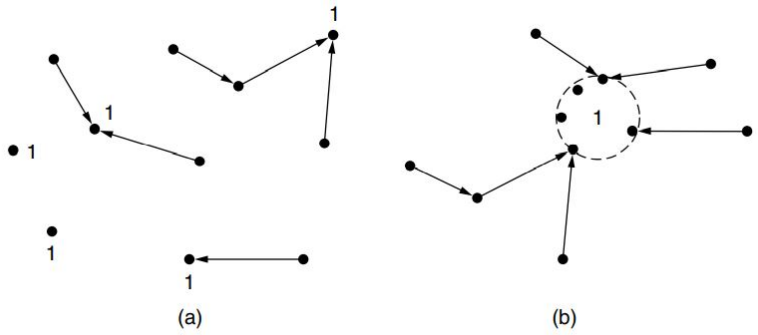


Figure 14: (a) Anycast routes to group 1. (b) Topology seen by the routing protocol.

Congestion Control

Too many packets present in the network causes packet delay and loss that degrades performance. This situation is called **congestion**. The network and transport layers share the responsibility for handling congestion. Since congestion occurs within the network, it is the network layer that directly experiences it and must ultimately determine what to do with the excess packets. However, the most effective way to control congestion is to reduce the load that the transport layer is placing on the network. This requires the network and transport layers to work together.

- Traffic-aware routing
- Admission control
- Traffic throtting
- Load shedding

Congestion results when too much traffic is offered; performance degrades due to loss/retransmissions

- Goodput (=useful packets) trails offered load

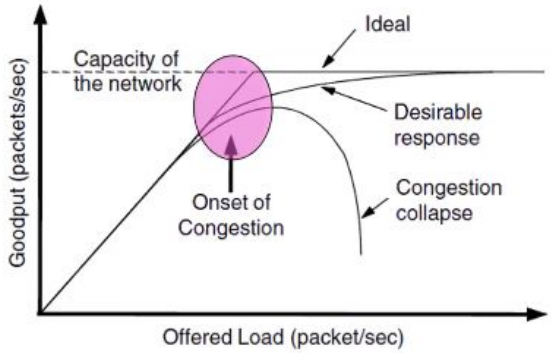


Figure 15: With too much traffic, performance drops sharply.

Congestion Control - Approaches

The presence of congestion means that the load is (temporarily) greater than the resources (in a part of the network) can handle. Two solutions come to mind:

Increase the resources or decrease the load. These solutions usually applied on different time scales to either prevent congestion or react to it once it has ocurred.

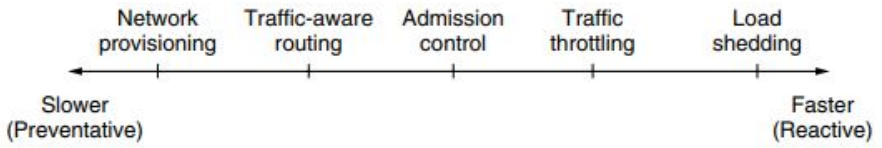


Figure 16: Timescales of approaches to congestion control

Traffic-Aware Routing

The goal in taking load into account when computing routes is to shift traffic away from hotspots that will be the first places in the network to experience congestion.

The most direct way to do this is to set the link weight to be a function of the (fixed) link bandwidth and propagation delay plus the (variable) measured load or average queuing delay. Least-weight paths will then favor paths that are more lightly loaded, all else being equal.

- E.g., use EI for West-to-East traffic if CF is loaded
- But take care to avoid oscillations

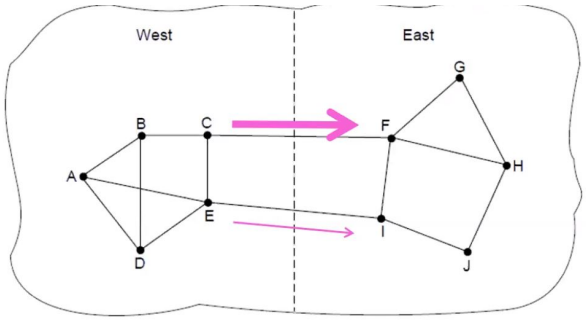


Figure 17: A network in which the East and West parts are connected by two links.

Admission Control

One technique that is widely used in virtual-circuit networks to keep congestion at bay is **admission control**. The idea is simple: do not set up a new virtual-circuit unless the network can carry the added traffic without becoming congested. Thus, attempts to set up a virtual circuit may fail. This is better than the alternative, as letting more people in when the network is busy just make matters worse. By analogy, in the telephone system, when a switch gets overloaded it practices admission control by not giving dial tones.

Traffic Throttling

In the Internet and many other computer networks, senders adjust their transmissions to send as much traffic as the network can readily deliver. In this setting, the network aims to operate just before the onset of congestion. When congestion is imminent, it must tell the senders to throttle back their transmission and slow down. This feedback is business as usual rather than an exceptional situation.

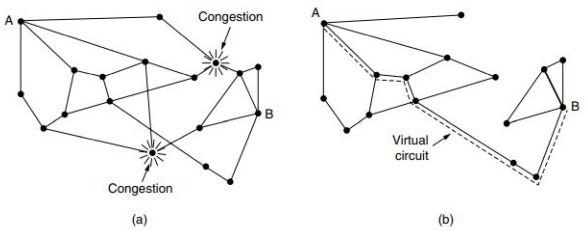


Figure 18: (a) A congested network. (b) The portion of the network that is not congested. A virtual circuit from A to B is also shown.

The term **congestion avoidance** is sometimes used to contrast this operating point with the one in which the network has become (overly) congested.

Congested routers signal hosts to slow down traffic

- ECN (Explicit Congestion Notification) marks packets and receiver returns signal to sender.

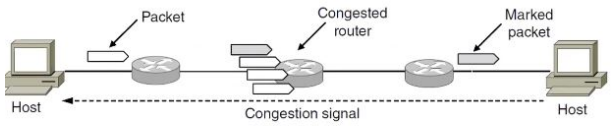


Figure 19: Explicit congestion notification

Load Shedding

When none of the above methods make the congestion disappear, routers can bring out the heavy artillery: load shedding. **Load shedding** is a fancy way of saying that when routers are being inundated by packets that they cannot handle, they just throw them away. The term comes from the world of electrical power generation, where it refers to the practice of utilities intentionally blacking out certain areas to save the entire grid from collapsing on hot summer days when the demand for electricity greatly exceeds the supply.

- When all else fails, network will drop packets (shed load).
- Can be done end-to-end or link-by-link.
- Link-by-link (right) produces rapid relief.

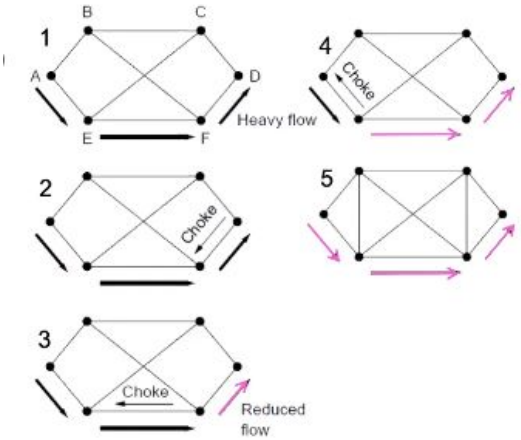


Figure 20: Link-by-link Load Shedding

Quality of Service

A stream of packets from a source to a destination is called a **flow**. A flow might be all the packets of a connection in a connection-oriented network, or all the packets sent from one process to another process in a connectionless network. The needs of each flow can be characterized by four primary parameters: bandwidth, delay, jitter, and loss. Together these deterine the **QoS (Quality of Service)** the flow requires.

Application requirements

Different applications care about different properties

Application	Bandwidth	Delay	Jitter	Loss
Email	Low	Low	Low	Medium
File sharing	High	Low	Low	Medium
Web access	Medium	Medium	Low	Medium
Remote login	Low	Medium	Medium	Medium
Audio on demand	Low	Low	High	Low
Video on demand	High	Low	High	Low
Telephony	Low	High	High	Low
Videoconferencing	High	High	High	Low

Table 1: Stringency of applications' quality-of-service requirements.

Network provides service with different kinds of QoS to meet application requirements.

Network Service	Application
Constant bit rate	Telephony
Real-time variable bit rate	Videoconferencing
Non-real-time variable bit rate	Streaming a movie
Available bit rate	File Transfer

Table 2: Example of QoS categories from ATM networks

Traffic Shaping

Before the network can make QoS guarantees, it must know what traffic is being guaranteed. In the telephone network, this characterization is simple. For example, a voice call(in uncompressed format) needs 64 kbps and consists of one 8-bit sample every 128μsec. However, traffic in data networks is **bursty**. It typically arrives at nonuniform rates as the traffic rate varies (e.g., videoconferencing with compression), users interact with applications (e.g., browsing a new Web page), and computers switch between tasks. Bursts of traffic are more difficult to handle than constant-rate traffic because they can fill buffers and cause packets to be lost.

Traffic shaping is a technique for regulating the average rate and burstiness of a flow of data that enters the network. The goal is to allow applications to transmit a wide variety of traffic that suits their needs, including some bursts, yet have a simple and useful way to describe the possible traffic patterns to the network.

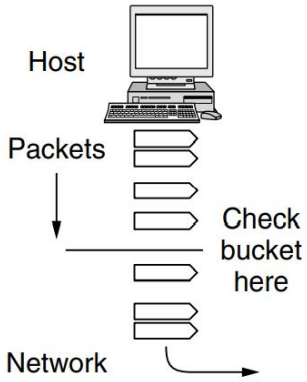


Figure 21: Shaping packets.

Token/Leaky bucket limits both the average rate (R) and short-term burst (B) of traffic

- For token, bucket size is B, water enters at rate R and is removed to send; opposite for leaky.

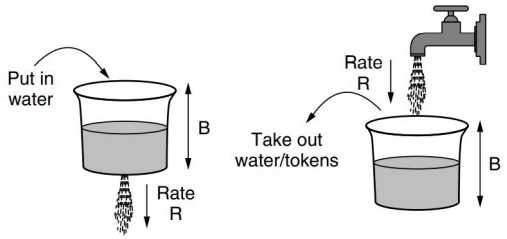


Figure 22: (a) A leaky bucket. (c) A token bucket.

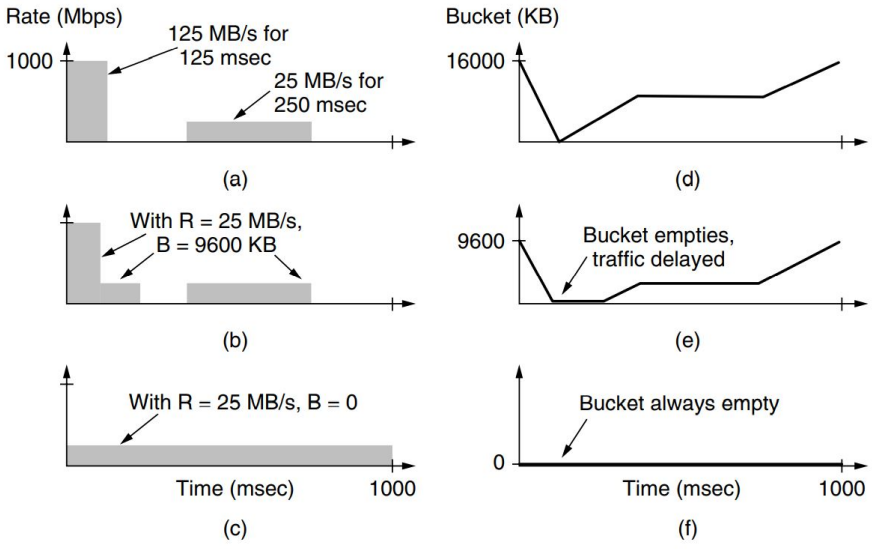


Figure 23: (a) Traffic from a host. Output shaped by a token bucket of rate 200 Mbps and capacity (b) 9600 KB and (c) 0 KB. Token bucket level of shaping with rate 200 Mbps and capacity (d) 16,000 KB, (e) 9600 KB, and (f) 0 KB.

Packet Scheduling

Being able to regulate the shape of the offered traffic is a good start. However, to provide a performance guarantee, we must reserve sufficient resources along the route that the packets take through the network. To do this, we are assuming that the packets of a flow follow the same route. Spraying them over routers at random makes it hard to guarantee anything. As a consequence, something similar to a virtual circuit has to be set up from the source to the destination, and all the packets that belong to the flow must follow this route.

Algorithms that allocate router resources among the packets of a flow and between competing flows are called **packet scheduling algorithms**. Three different kinds of resources can potentially be reserved for different flows:

1. Bandwidth
2. Buffer space
3. CPU cycles

Packet scheduing divides router/link resources among traffic flows with alternatives to FIFO (First In First Out).

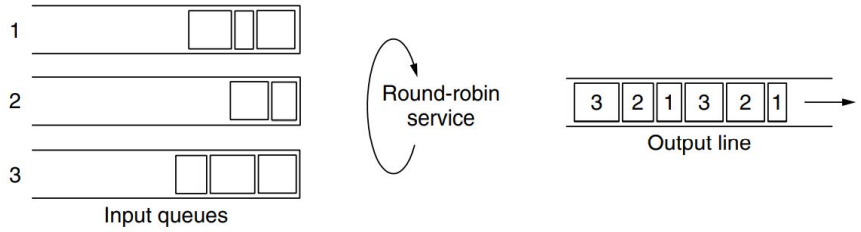
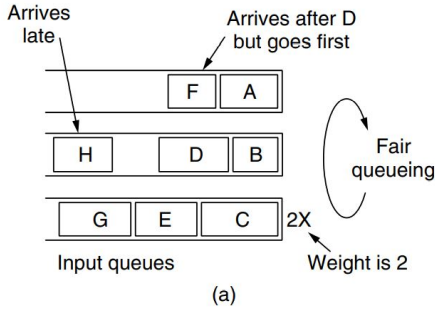


Figure 24: Round-robin fair queueing.

Fair Queueing approximates bit-level fairness with different packet sizes; weights change target levels

- Result is WFQ (Weighted Fair Queueing)



Packet	Arrival time	Length	Finish time	Output order
A	0	8	8	1
B	5	6	11	3
C	5	10	10	2
D	8	9	20	7
E	8	8	14	4
F	10	6	16	5
G	11	10	19	6
H	20	8	28	8

(b)

Figure 25: (a) Weighted Fair Queueing. (b) Finishing times for the packets.

Admission Control

Admission control takes a traffic flow specification and decides whether the network can carry it.

- Sets up packet scheduling to meet QoS

Parameter	Unit
Token bucket rate	Bytes/sec
Token bucket size	Bytes
Peak data rate	Bytes/sec
Minimum packet size	Bytes
Maximum packet size	Bytes

Table 3: An example flow specification.

One method of relating flow specification to router resources that correspond to bandwidth and deolay performance guarantees is given by Parekh and Gallagher. It is based on traffic sources shaped by (R, B) token buckets and WFQ at routers. Each flow is given a WFQ weight W large enough to drain its token bucket rate R as shown in below.

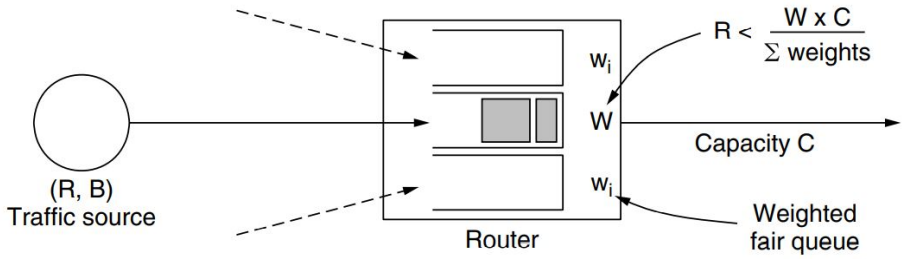


Figure 26: Bandwidth and delay guarantees with token buckets and WFQ.

- Shape traffic source to a (R, B) token bucket
- Run WFQ with weight W / all weights > R/capacity
- Holds for all traffic patterns, all topologies

Integrated Services

Integrated services was aimed at both unicast and multicast applications. An example of the former is a single user streaming a video clip from a news site. An example of the latter is a collection of digital television stations broadcasting their programs as streams of IP packets to many receivers at various locations.

Design with QoS for each flow; handles multicast traffic.

Admission with RSVP (Resource reSerVation Protocol):

- Receiver sends a request back to the sender
- Each router along the way reserves resources
- Routers merge multiple requests for same flow
- Entire path is set up, or reservation not made

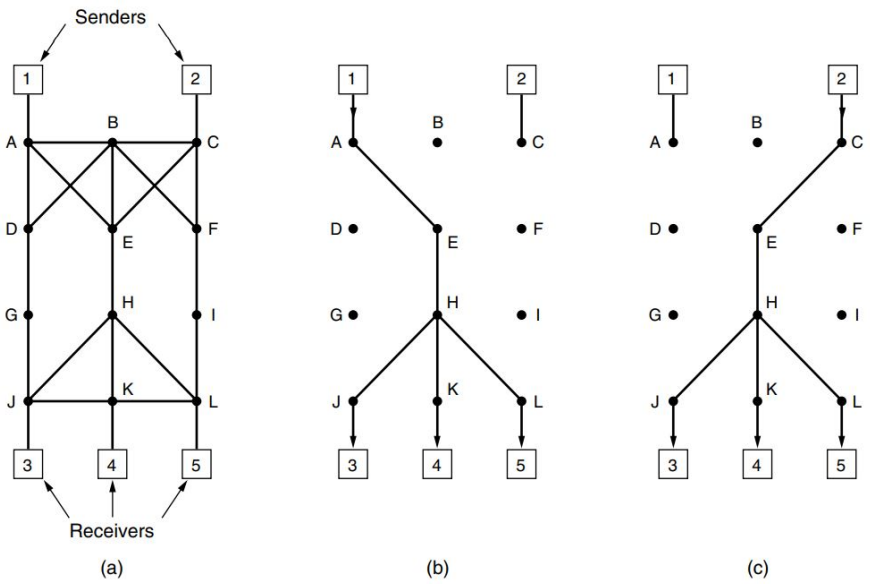


Figure 27: (a) A network. (b) The multicast spanning tree for host 1. (c) The multicast spanning tree for host 2.

Differentiated Services

Design with classes of QoS; customers buy what they want

- Expedited class is sent in preference to regular class
- Less expedited traffic but better quality for applications

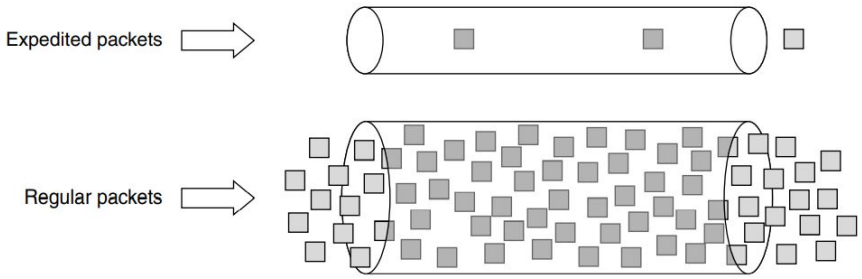


Figure 28: Expedited packets experience a traffic-free network.

Implementation of DiffServ:

- Customers mark desired class on packet
- ISP shapes traffic to ensure markings are paid for
- Routers use WFQ to give different service levels

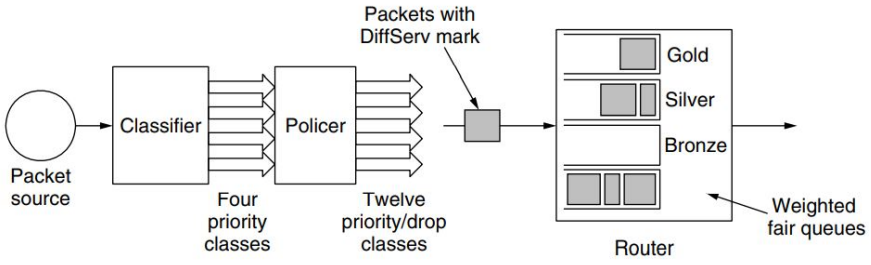


Figure 29: A possible implementation of assured forwarding.

Internetworking

Until now, we have implicitly assumed that there is a single homogeneous network, with each machine using the same protocol in each layer. Unfortunately, this assumption is wildly optimistic. Many different networks exits, including PANs, LANs, MANs, and WANs. We have described Ethernet, Internet over cable, the fixed and mobile telephone networks, 802.11, 802.16, and more. Numerous protocols are in widespread use across these networks in every layer. We will take a look at the issues that arise when two or more networks are connected to form an **internetwork**, or more simply an **internet**.

How Networks Differ

Networks can differ in many ways. Some of the differences, such as different modulation techniques or frame formats, are internal to the physical and data link layers. These differences will not concern us here. Instead, the list below some of the differences that can be exposed to the network layer.

Item	Some Possibilities
Service offered	Connectionless versus connection oriented
Addressing	Different sizes, flat or hierarchical
Broadcasting	Present or absent (also multicast)
Packet size	Every network has its own maximum
Ordering	Ordered and unordered delivery
Quality	Present or absent; many different kinds
Reliability	Different levels of loss
Security	Privacy rules, encryption, etc.
Parameters	Different timeouts, flow specifications, etc.
Accounting	By connect time, packet, byte, or not at all

Table 4: Some of the many ways networks can differ.

How Networks Can Be Connected

There are two basic choices for connecting different networks: we can build devices that translate or convert packets from each kind of network into packets for each other network, or we can try to solve the problem by adding a layer of indirection and building a common layer on top of the different networks.

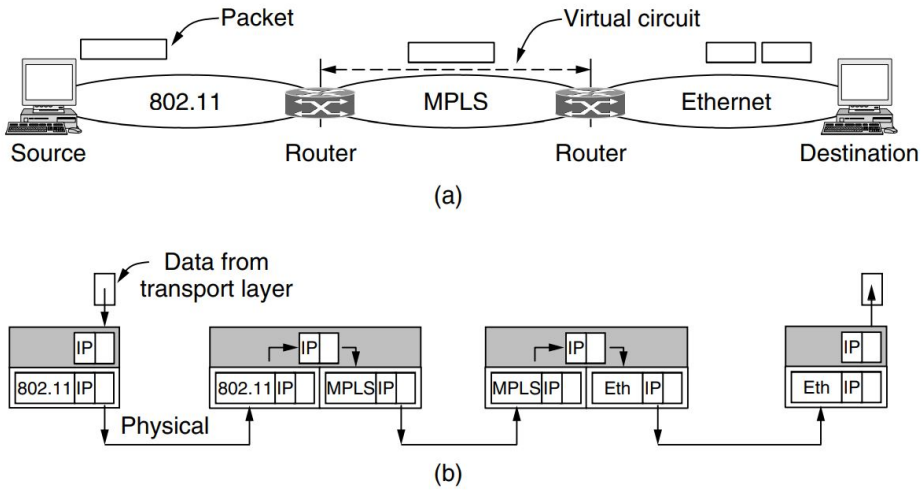


Figure 30: (a) A Packet crossing different networks. (b) Network and link layer protocol processing.

Tunneling

Handling the general case of making two different networks interwork is exceedingly difficult. However, there is a common special case that is manageable even for different network protocols. This case is where the source and destination hosts are on the same type of network, but there is a different network in between. As an example, think of an international bank with an IPv6 network in Paris, an IPv6 network in London and connectivity between the offices via the IPv4 Internet.

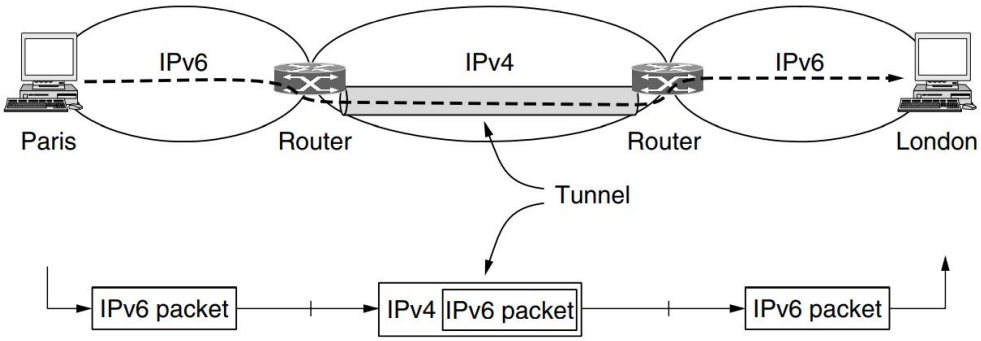


Figure 31: Tunneling a packet from Paris to London.

Packet Fragmentation

Each network or link imposes some maximum size on its packets. These limits have various causes, among them

1. Hardware (e.g., the size of an Ethernet frame)
2. Operating System /e.g., all buffers are 512 bytes).
3. Protocols (e.g., the number of bits in the packet length field).
4. Compliance with some (inter)national standard.
5. Desire to reduce error-induced retransmissions to some level.

6. Desire to prevent one packet from occupying the channel too long.

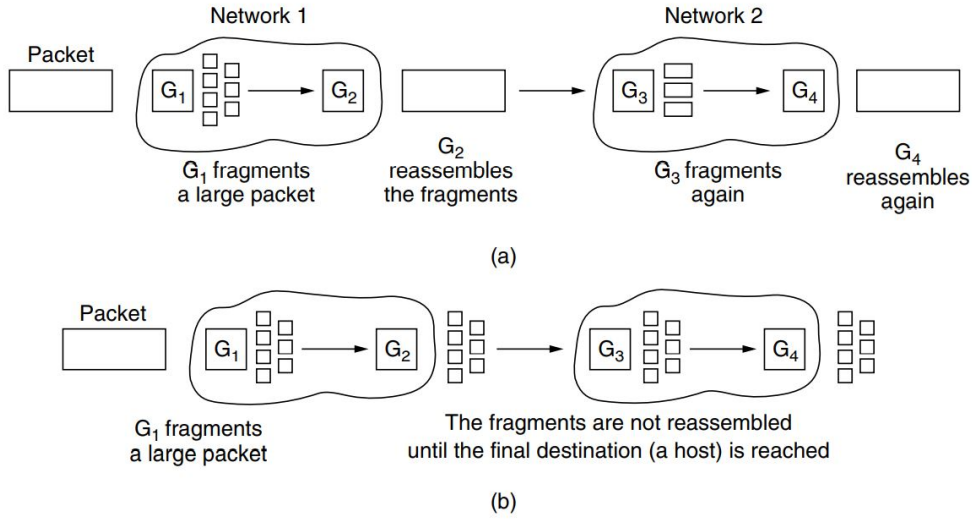


Figure 32: (a) Transparent fragmentation. (b) Nontransparent fragmentation.

Network Layer in the Internet

Internet is an interconnected collection of many networks that is held together by the IP protocol.

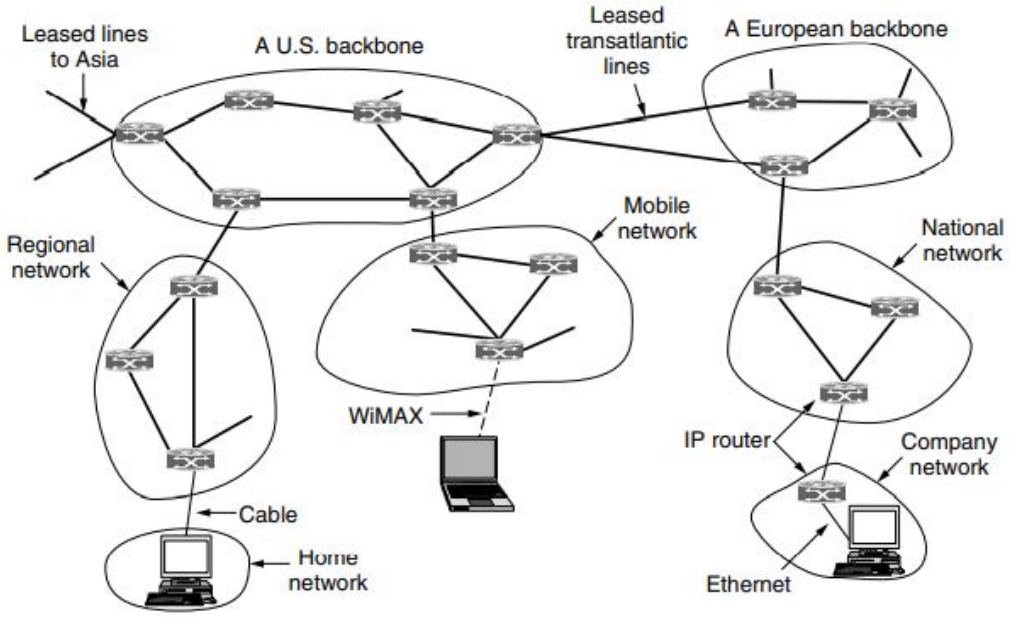


Figure 33: The Internet is an interconnected collection of many networks.

The IP Version 4 Protocol

An appropriate place to start study of the network layer in the Internet is the format of the IP datagrams themselves. An IPv4 datagram consists of a header part and a body or payload part. The header has a 20-byte fixed part and a variable-length optinoal part.

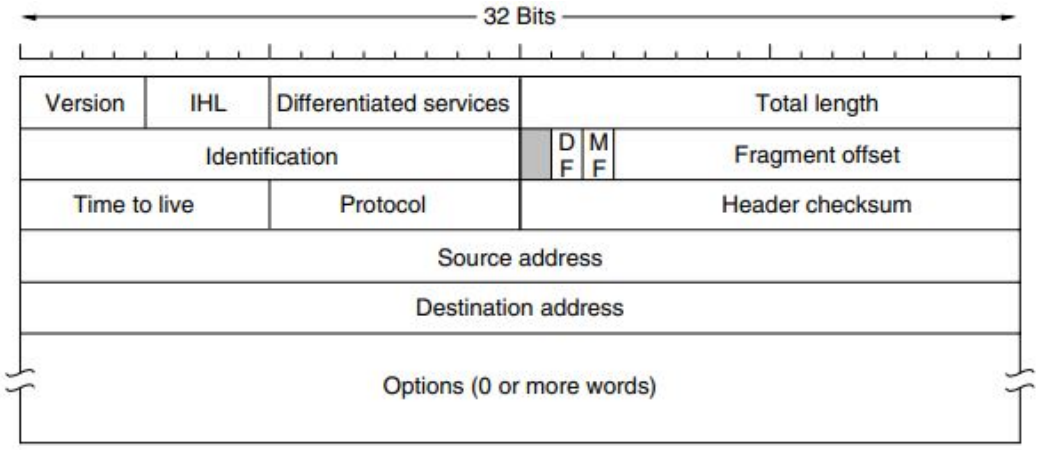


Figure 34: The IPv4 (Internet Protocol) header.

The Version field keeps track of which version of the protocol the datagram belongs to. Version 4 dominates the Internet today. By including the version at the start of each datagram, it becomes possible to have a transition between versions over a long period of time. In fact, IPv6, the next version of IP, was defined more than a decade ago, yet is only just beginning to be deployed.

IP Addresses

A defining feature of IPv4 is its 32-bit addresses. Every host and router on the Internet has an IP address that can be used in the *Source address* and *Destination address* fields of IP packets. It is important to note that an IP address does not actually refer to a host. It really refers to a network interface, so if a host is on two networks, it must have two IP address. However, in practice, most hosts are on one network and thus have one IP address. In contrast, routers have multiple interface and thus multiple IP addresses.

IP Addresses - Prefixes

IP addresses are hierarchical, unlike Ethernet addresses. Each 32-bit address is comprised of a variable-length network portion in the top bits and a host portion in the bottom bits. The network portion has the same value for all hosts on a single network, such as an Ethernet LAN. This mean that a network corresponds to a contiguous block of IP address space. This block is called a **prefix**.

Since the prefix length cannot be inferred from the IP address alone, routing protocols must carry the prefixes to routers. Sometimes prefixes are simply described by their length, as in a "/16" which is pronounced "slash 16." The length of the prefix corresponds to a binary mask of 1s in the network portion. When written out this way, it is called a **subnet mask.** It can be ANDed with the IP addres to extract only the network portion. For the example, the subnet mask is 255.255.255.0.

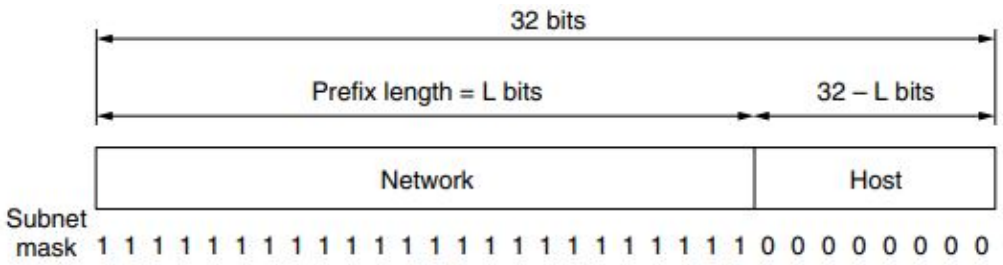


Figure 35: An IP prefix and a subnet mask.

IP Addresses - Classful Addressing

Old addresses came in blocks of fixed size (A, B, C)

- Carries size as part of addresses, but lacks flexibility
- Called classful (vs. classless) addressing

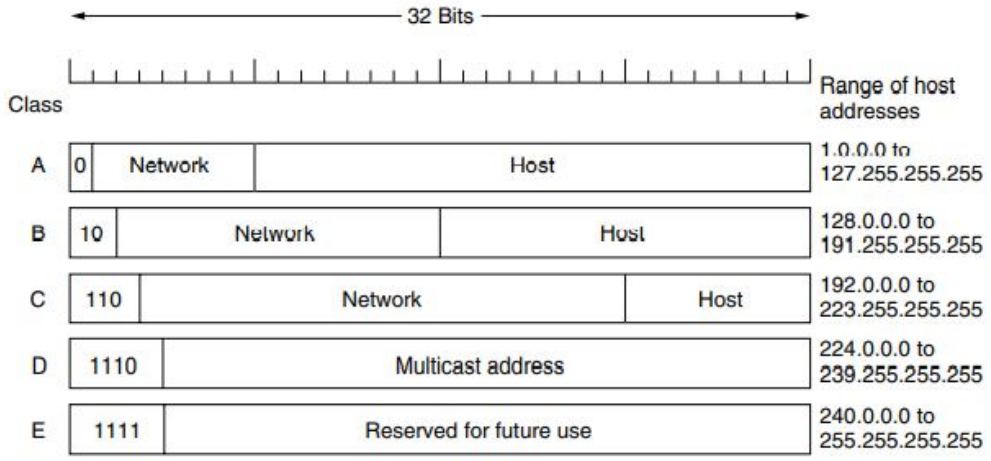


Figure 36: IP address formats.

IP Addresses - Aggregation

Aggregation joins multiple IP prefixes into a single larger prefix to reduce routing table size.

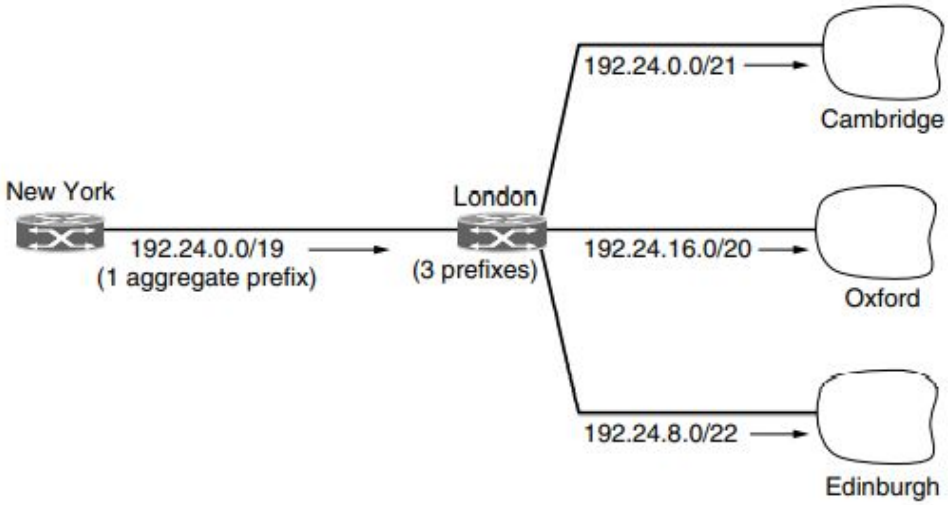


Figure 37: Aggregation of IP prefixes.

IP Addresses - NAT

IP addresses are scarce. An ISP might have a /16 address, giving it 65,534 usable host numbers. If it has more customers than that, it has a problem.

NAT (Network Address Translation) box maps one external IP addrss to many internal IP addresses.

- Uses TCP/UDP port to tell connections apart
- Violates layering; very common in homes, etc.

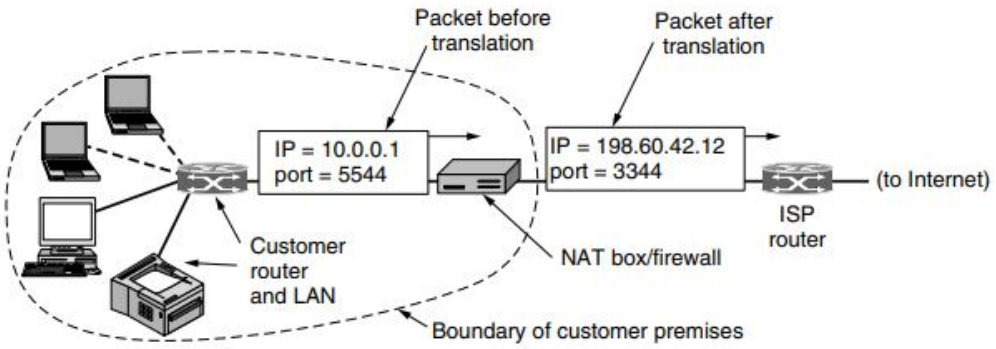


Figure 38: Placement and operation of a NAT box.

IP Version 6

IP has been in heave use for decades. It has worked extremely well, as demonstrated by the exponential growth of the Internet. Unfortunately, IP has become a victim of its own popularity: it is close to running out of addresses. Even with CIDR and NAT using addresses more sparingly, the last IPv4 addresses are expected to be assigned by ICANN before the end of 2012.

The only long-term solution is to move to larger addresses. **IPv6** is replacement design that does just that. It uses 128-bit addresses; a shortage of these addresses is not likely any time in the foreseeable future. However, IPv6 has proved very difficult to deploy. It is a different network layer protocol that does not really interwork with IPv4, despite many similarities. Also companies and users are not really sure why they should want IPv6 in any case.

Major upgrade in the 1990s due to impending address exhaustion, with various other goals:

- Support billions of hosts
- Reduce routing table size
- Simplify protocol
- Better security
- Attention to type of service
- Aid multicasting
- Roaming host without changing address
- Allow future protocol evolution
- Permit coexistence of old, new protocols, ...

Deployment has been slow & painful, but may pick up pace now that addresses are all but exhausted.

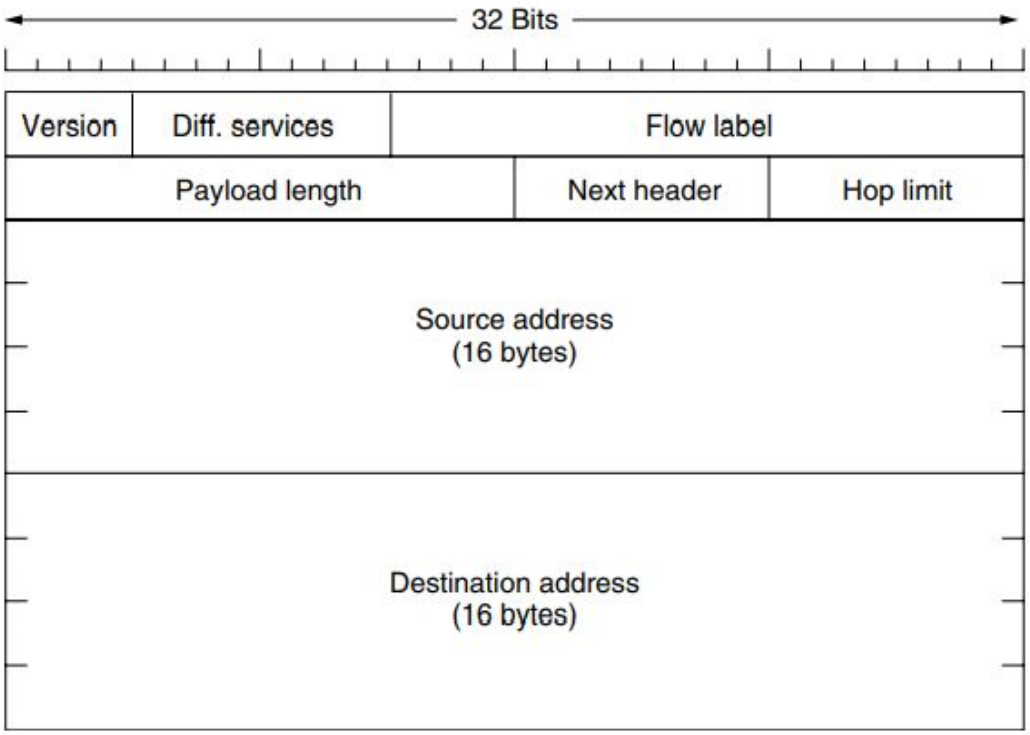


Figure 39: The IPv6 fixed header (required).

- IPv6 extension headers handles other functionality.

Extension header	Description
Hop-by-hop options	Miscellaneous information for routers
Destination options	Additional information for the destination
Routing	Loose list of routers to visit
Fragmentation	Management of datagram fragments
Authentication	Verification of the sender's identity
Encrypted security payload	Information about the encrypted contents

Table 5: IPv6 extension headers

Internet Control Protocols

In addition to IP, which is used for data transfer, the Internet has several companion control protocols that are used in the network layer. They include ICMP, ARP, and DHCP.

- ICMP is a companion to IP that returns error info
  - Required, and used in amny ways, e.g., for traceroute
- ARP finds Ethernet address of a local IP address
  - Glue that is needed to send any IP packets
  - Host queries an address and the owner replies
- DHCP assigns a local IP address to a host
  - Gets host started by automatically configuring it
  - Host sends request to server, which grants a lease

Main ICMP (Internet Control Message PRotocol) types:

Message type	Description
Destination unreachable	Packet could not be delivered
Time exceeded	Time to live field hit 0
Parameter problem	Invalid header field
Source quench	Choke packet
Redirect	Teach a router about geography
Echo and echo reply	Check if a machine is alive
Timestamp request/reply	Same as Echo, but with timestamp
Router advertisement/solicitation	Find a nearby router

Table 6: The principal ICMP message types.

ARP (Address Resolution Protocol) lets nodes find target Ethernet addresses [purple] from their IP addresses

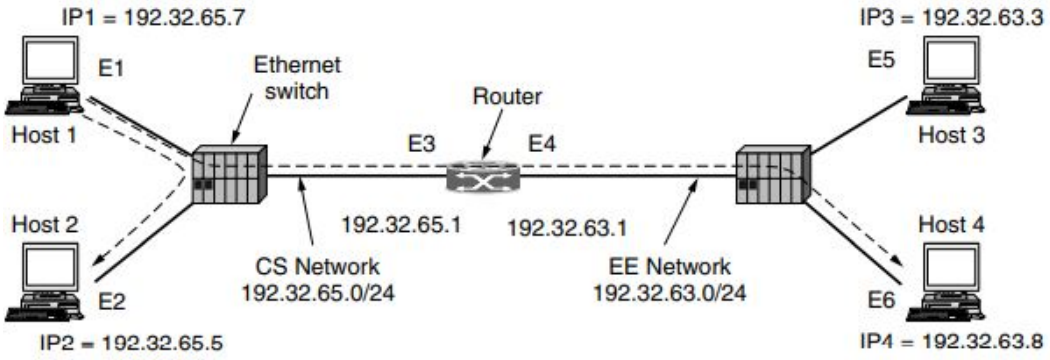


Figure 40:

Frame	Source IP	Source Eth.	Destination IP	Destination Eth.
Host 1 to 2, on CS net	IP1	E1	IP2	E2
Host 1 to 4, on CS net	IP1	E1	IP4	E3
Host 1 to 4, on EE net	IP1	E4	IP4	E6

Table 7: Two switched Ethernet LANs joined by a router.

Label Switching and MPLS

There is a kind of technology that is starting to be widely used, especially by ISPs, in order to move Internet traffic across their network. This technology is called **MPLS (MultiProtocol Label Switching)** and it is perilously close to circuit switching.

MPLS sends packets along established paths; ISPs can use for QoS

- Path indicated with label below the IP layer

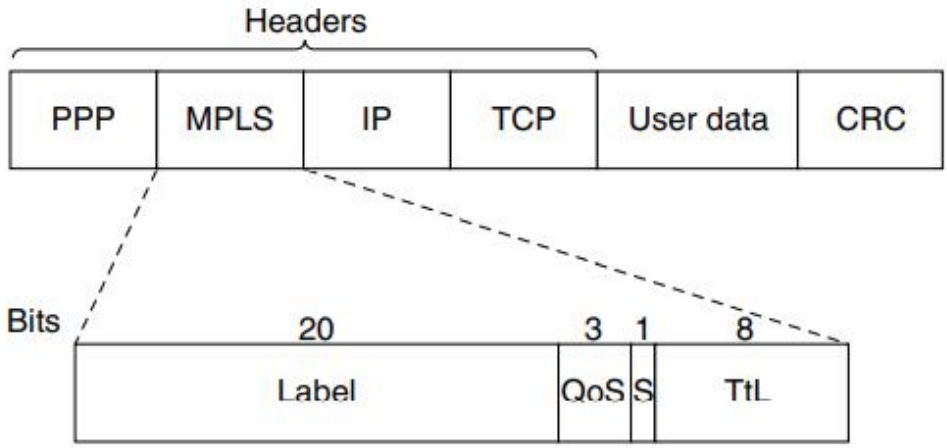


Figure 41: Transmitting a TCP segment using IP, MPLS, and PPP.

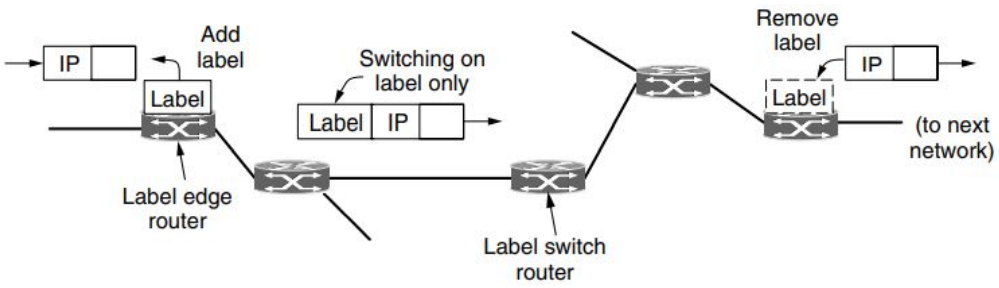


Figure 42: Forwarding an IP packet through an MPLS network.

OSPF - Interior Routing Protocol

OSPF computes routes for a single network (e.g., ISP)

- Models network as a graph of weighted edges

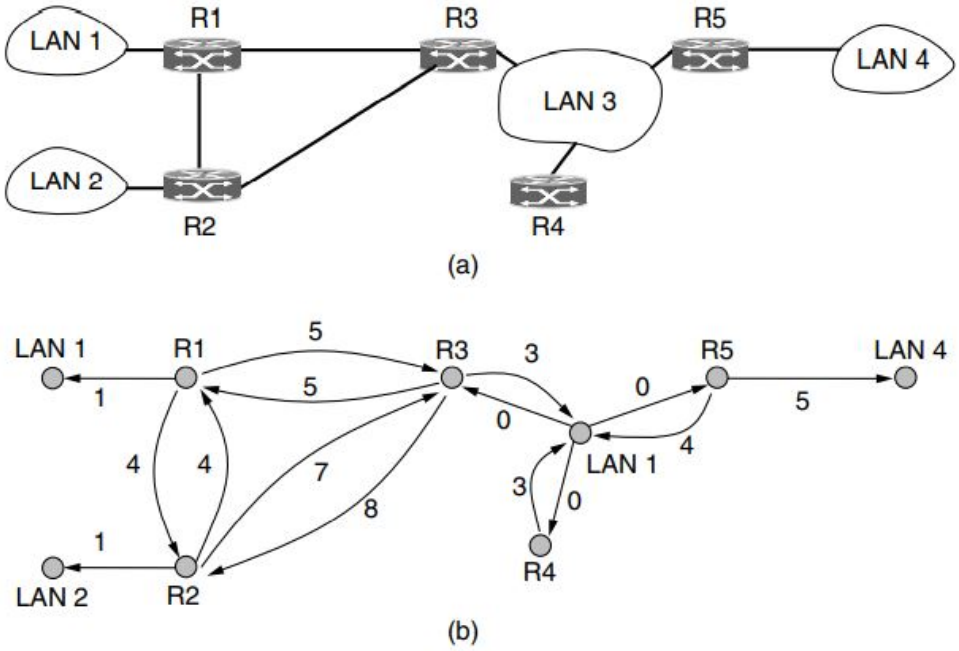


Figure 43: (a) An autonomous system. (b) A graph representation of (a).

OSPF divides one large network (Autonomous System) into areas connected to a backbone area

- Helps to scale; summaries go over area borders

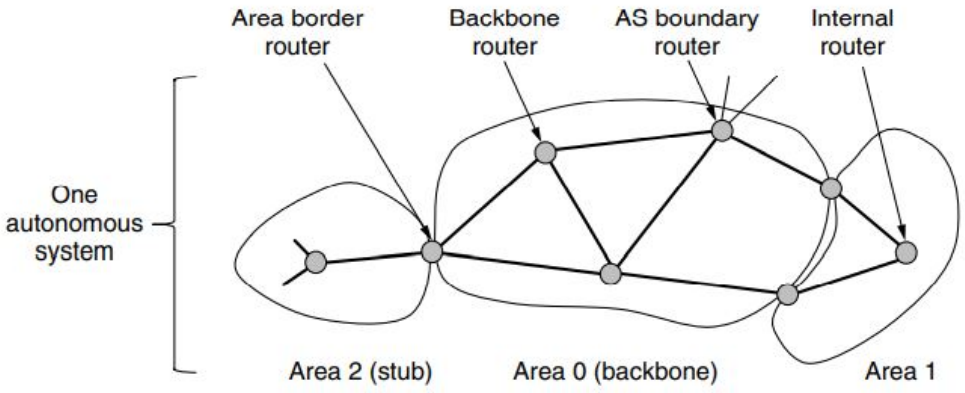


Figure 44: The relation between ASes, backbones, and areas in OSPF.

BGP - Exterior Routing Protocol

BGP (Border Gateway Protocol) computes routes across interconnected, autonomous network

- Key role is to respect networks' policy constraints

Common policy distinction is transit vs. peering:

- Transit carries traffic for pay; peers for mutual benefit
- AS1 carries AS2↔ AS4 (Transit) but not AS3 (Peer)

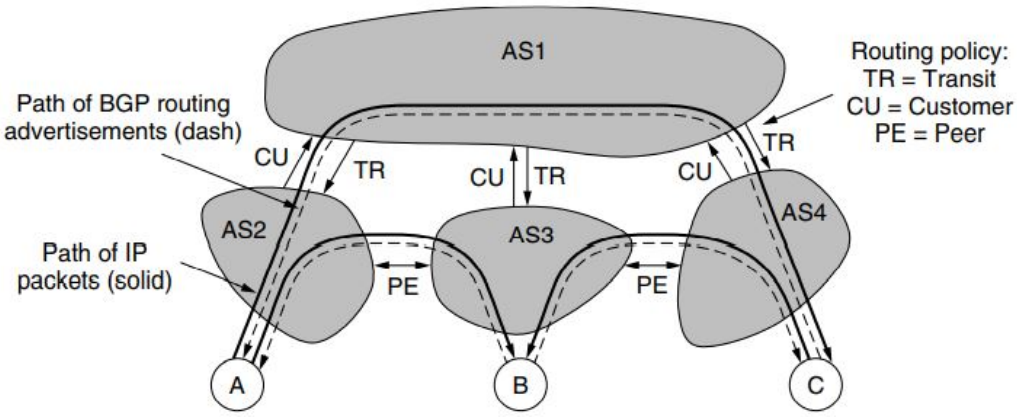


Figure 45: Routing policies between four autonomous systems.

BGP propagates messages along policy-compliant routes

- Message has prefix, AS path (to detect loops) and nexthop IP (to send over the local network)

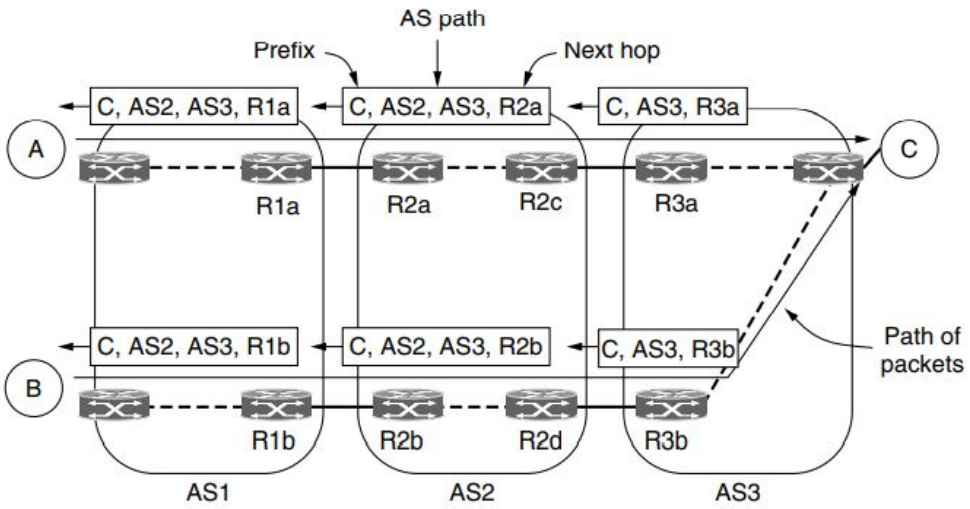


Figure 46: Propagation of BGP route advertisements.

Internet Multicast

Normal IP communication is between one sender and one receiver. However, for some applications, it is useful for a process to be able to send to a large number of receivers simultaneously. Examples are streaming a live sports event to many viewers, delivering program updates to a pool of replicated servers, and handling digital conference telephone calls.

Groups have a reserved IP address range (class D)

- Membership in a group handled by IGMP (Internet Group Management Protocol) that runs at routers

Routers computed by protocols such as PIM:

- Dense mode uses RPF with pruning
- Sparse mode uses core-based trees

IP multicasting is not widely used except within a single network, e.g., datacenter, cable TV network.

Mobile IP

Many users of the Internet have mobile computers and want to stay connected when they are away from home and even on the road in between. Unfortunately, the IP addressing system makes working far from home easier said than done, as we will describe shortly. When people began demanding the ability anyway, IETF set up a Working Group to find a solution. The Working Group quickly formulated a number of goals considered desirable in any solution. The major ones were:

1. Each mobile host must be able to use its home IP address anywhere.
2. Software changes to the fixed hosts were not permitted.
3. Changes to the router software and tables were not permitted.

4. Most packets for mobile hosts should not make detours on the way.
5. No overhead should be incurred when a mobile host is at home.

- Home agent tunnels packets to reach the mobile host; reply can optimize path for subsequent packets
- No changes to routers or fixed hosts

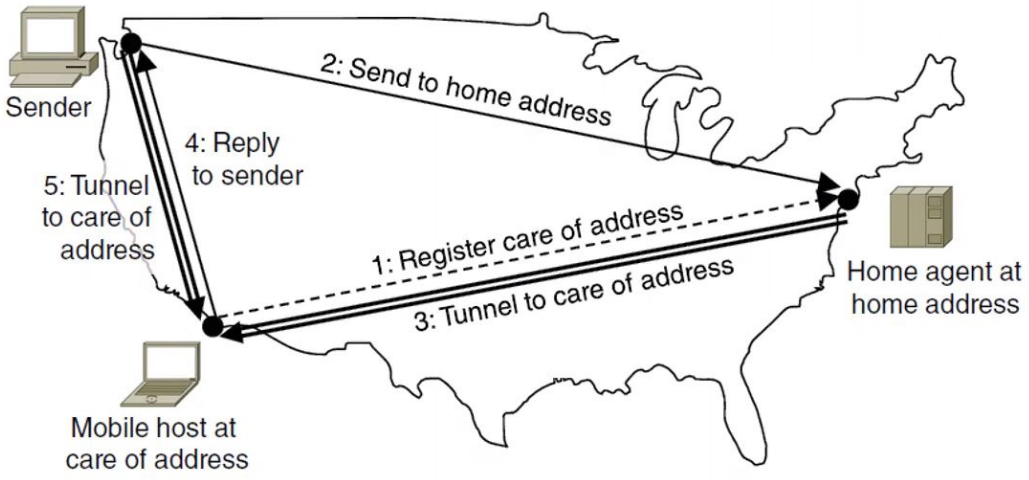


Figure 47: